

PCT

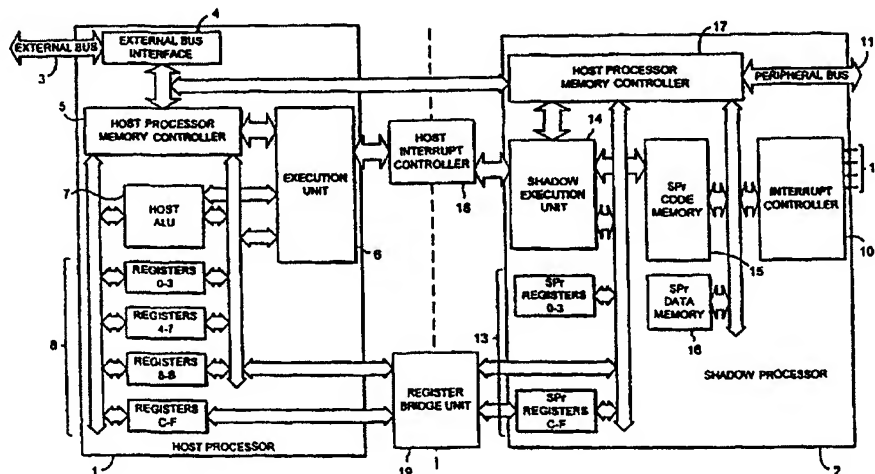
WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁷ : G06F 9/38		A1	(11) International Publication Number: WO 00/26772
			(43) International Publication Date: 11 May 2000 (11.05.00)
(21) International Application Number: PCT/EP99/08058		(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 25 October 1999 (25.10.99)			
(30) Priority Data: 9823819.9 30 October 1998 (30.10.98) GB			
(71) Applicant (for all designated States except US): TELEFONAKTIEBOLAGET LM ERICSSON [SE/SE]; S-126 25 Stockholm (SE).			
(72) Inventor; and (75) Inventor/Applicant (for US only): NAYLOR, Rowan, Nigel [GB/GB]; 4 Lisa Court, Downsland Road, Basingstoke RG21 1TU (GB).			
(74) Agent: VIGARS, Christopher, Ian; Haseltine Lake & Co., Imperial House, 15-19 Kingsway, London WC2B 6UD (GB).		Published With international search report.	

(54) Title: PROCESSING ARRANGEMENTS



(57) Abstract

A processing arrangement for a computer comprising: first processor means (1) for processing a first set of instructions; and second processor means (2) for processing a second set of instructions, the second set of instructions being a subset of the first set of instructions, wherein the second processor means (2) is arranged to receive control signals and to process instructions in dependence upon those control signals without reference to the first processor means.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NE	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NL	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NO	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NZ	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PL	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakhstan	RO	Romania		
CU	Cuba	LC	Saint Lucia	RU	Russian Federation		
CZ	Czech Republic	LI	Liechtenstein	SD	Sudan		
DE	Germany	LK	Sri Lanka	SE	Sweden		
DK	Denmark	LR	Liberia	SG	Singapore		
EE	Estonia						

PROCESSING ARRANGEMENTS

The present invention relates to processing arrangements for computer architectures.

DESCRIPTION OF THE RELATED ART

5 There are many conventional computer architectures which are based on a single main processor cooperating with a co-processor. The co-processor adds functionality that the main processor in the architecture does not have or does not perform particularly efficiently. The co-processor generally
10 uses instructions which are not implemented in the instruction set of the main processor. As such, many co-processors are used to address very specific code requirements, for example floating point arithmetic or
15 signal processing. In most applications, this means that the instruction set of the co-processor is specific to that co-processor.

 In addition, many main processors use real time operating systems to service multiple tasks and
20 exceptions, such as interrupts. Servicing multiple tasks can result in context changes which can absorb significant amounts of the processing power available in the processor. A context change occurs when the task being executed by a processor is changed. The
25 context of a task relates to the code corresponding to the task, and the state of the internal registers of the processor. Furthermore, in low power applications, in order to conserve power, sleep modes are used from which the processor must be reactivated when interrupt
30 or service requests occur. When the processor is reactivated the context must be loaded and then the service performed, the processor then returns to an inactive state. Such a process can consume a large amount of power.

35 It is therefore desirable to provide a computer

architecture which can overcome these disadvantages.

SUMMARY OF THE PRESENT INVENTION

According to one aspect of the present invention there is provided a processing arrangement for a computer comprising:

first processor means for processing a first set of instructions; and

second processor means for processing a second set of instructions, the second set of instructions being a subset of the first set of instructions, wherein the second processor means is arranged to receive control signals and to process instructions in dependence upon those control signals without reference to the first processor means.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram illustrating a computer processor architecture in accordance with the present invention;

Figure 2 shows a computer architecture including multiple processors; and

Figure 3 is a diagrammatic illustration of how processors embodying the present invention can realise the functionality of a desired virtual processor.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Figure 1 shows a block diagram illustrating a host (or first) processor 1 connected with a shadow (or second) processor 2. In the example shown, the shadow processor 2 is used to control interrupts received from peripherals connected to the processor system.

The host processor 1 communicates with an external bus 3 by way of an external bus interface 4. The external bus 3 is used for transferring data to and from the main processor and memory devices (not shown).

The host processor 1 also includes a memory controller 5 for controlling data access with memory devices. The memory controller 5 is controlled itself by an execution unit 6 which has overall control of the main processor 1. The host processor 1 also include an arithmetic logic unit (ALU) 7 and a number of registers 8, sixteen in the example shown. The various components of the host processor 1 communicate with each other by way of appropriate internal buses.

10 In order that the host processor 1 can be held in a low power inactive mode for as long as possible, a shadow processor 2 is connected to the main processor 1. The shadow processor 2 includes an interrupt controller 10 which receives interrupt signals via interrupt inputs 12. The shadow processor also includes registers which correspond to selected ones of the registers of the host processor 1. The interrupt controller can be connected, as in the example of Figure 1, to peripherals connected to a peripheral bus 11.

20 The shadow processor 2 also includes an execution unit 14, code memory 15, and data memory 16. The shadow processor 2 operates to process a selected subset of the instructions from which the host processor operates, and these instructions are stored in the code memory 15. The shadow processor 2 has access to the host processor memory by way of a memory controller 17 which interfaces with the external bus 3 by way of the host external bus interface 4.

30 In order to provide a link between the host processor 1 and the shadow processor 2, the execution units of the two processors are connected by way of a host interrupt controller 18, and the registers of the processors are connected by a register bridge unit 19.

35 The Host Interrupt Controller 18 is a module which

-4-

allows the shadow processor to issue an interrupt to the host processor in order to cause a change in task execution, a context change, appropriate to the requirements of the system. It generates an interrupt to the host using the host interrupt protocol and indicating the source of the interrupt as a vector programmed by the shadow process related to the new task required.

The following description, of the process used to service an interrupt received from a peripheral bus 11 is one example of the operation of the shadow processor 2. A peripheral (not shown) raises an interrupt request on the interrupt input lines 12. The input controller 10 operates to interpret the interrupt request, and if the interrupt request is of the type able to be processed by the shadow processor 2, then the shadow processor 2 will service that interrupt. An example of an interrupt is if an external device such as a serial port has received data from a system to which it is attached. The data from such a serial port may be contained within a message body including information about the source and content of the data. The shadow processor 2 then services the interrupts required by receiving the message then removing the data from that message. Once the data has been removed and checked it issues an interrupt to the main processor 1 via the Host Interrupt Controller 18 having first set up the context needed by the task to service the data within the message. The main processor 1 would begin processing the data while the shadow processor 2 stores the context of the interrupted task.

The Register Bridge unit 19 allows multiple access to a set of registers by both host and shadow processor. It allows physical registers to appear in both processor systems while resolving any conflict in

their access and mapping of those registers within the physical resource space of the processors.

5 This unit contains multiplexers to allow dual access to either processor buses, arbitration logic to prevent access conflict, and logic to allow access addresses to be altered to allow re-mapping of the location of the registers in the address map of either processor.

10 For example registers mapped to positions 4-7 in the host may appear also in positions 4-7 in the shadow or be re-mapped to 8-11 of the shadow. Or in the case of register swapping, registers x-y of the host may be exchanged for equivalent registers in the shadow and vice-versa. This allows registers x-y of the shadow to be loaded with a new context then swapped in to the host in a context change. The registers replaced in the shadow would contain the old host context and can be stored or re-stored by the shadow while the host is processing the new context.

20 While the host processor 1 is processing a task or is in an unactive ("sleep") mode, the shadow processor 2 will accept interrupts to determine the requirement and decide on the action required. If the action can be handled by the shadow processor alone then it is serviced without reference to the host, but if it does require some intervention by the host, the shadow processor 2 will activate the host processor by way of the host interrupt controller 18.

30 In the example shown, the shadow processor 2 can service routine requests from the peripherals without the need for host intervention. Data to be processed is shared between the host processor and the shadow processor by way of the register bridge unit 19, in order that the same data can be operated on by both processors.

35

It will be readily appreciated that, the shadow processor 2 is able to use a subset of the instruction set of the host processor. The main advantage of such a shadow processor is that instruction code can be readily shared between the host and the shadow, without the need for processor emulation or complex code conversion. The other significant advantage is that any work needed to develop the shadow processor can be based on the host processor development work. In addition, the shadow processor design can be optimised, based specifically on the processing of a limited range of instructions of the host processor. Code written for such an application can be analysed to identify the most frequently used instructions, and shadow processors can be provided for those most frequently used instructions. The shadow processor can then be based on the common instruction set but minimum register and addressing range requirements. The shadow processor has access to all the relevant register and memory areas used by the host processor and so the host and shadow processor can work in tandem, the shadow processor performing tasks for which it is designed independently of any reference to the host processor. It will be appreciated that a plurality of shadow processors can be used with a single host processor to provide efficient data processing without the need to refer directly to the host processor itself.

One particular advantage of such system is that the shadow processor can service interrupt requests and other routine tasks without the need to move the host processor into an active state. This can reduce the power consumption of the processing system in total.

It will be appreciated that it is possible to provide all of the functionality of a single host processor by a plurality of shadow processors each

dedicated to a single part of that functionality of the host processor. In this way it is possible to produce a processing architecture which does not need to use a host processor as such, but can use a group of shadow processors each of which is optimised for a particular function. Such architectures are illustrated in Figures 2 and 3, which show that shadow processors can be arranged to share data and registers so that processing of program steps can have continuity.

Figure 2 is an application of a shadow processor concept without the use of a host processor.

A number of shadow processors are arranged in sequence; each processor performs a particular process on data derived from previous shadow processors and passes the processed data on to the next in the sequence. The sequence has input/output ports on the end shadow processors for connection to other system components. Processing occurs in both directions from shadow processor (i.e. 1 to 5 and 5 to 1). In addition sections 3 and 4 contain two shadow processors (A&B) with additional input/output to be combined with the main sequence. Shadow processor 6 performs the task of memory control and ensures that appropriate code is loaded into the shadow processors from main memory as required. It also performs the main co-ordination task for the system.

An application is of such a configuration could be a telecommunications terminal. The I/O of shadow processor 1 would be attached to the RF subsystem, I/O of processor 5 to the Audio sub-system. The secondary I/O of shadow processors 3b/4b to the data processing, and user interfaces e.g. LCD and keyboard. Each of the shadow processors 1-5 perform specific tasks required such as audio codec, interleaving/de-interleaving, etc.

Figure 3 is an example of a virtual processor

concept.

The main processor is provided as a virtual processor, which may only exist in concept and as a simulation. From this virtual processor various applications are analysed and specific implementations derived. Three such implementations 1-3 are drawn below the Host. Each of the implementations contains specific components of the host required by the applications. For example, Shadow processor A. (1) could be characteristic of a RISC CPU, Shadow processor B (2) could be an example of a logic processor, and Shadow processor C (3) could be an example of a memory management processor.

The important distinction of figure 3 is that the host is a virtual processor it may not physically exist.

The shadow processors are optimised fragments of the virtual processor synthesised to a physical implementation on a chip.

CLAIMS

1. A processing arrangement for a computer comprising:

5 first processor means for processing a first set of instructions; and

second processor means for processing a second set of instructions, the second set of instructions being a subset of the first set of instructions, wherein the second processor means is arranged to receive control
10 signals and to process instructions in dependence upon those control signals without reference to the first processor means.

2. An arrangement as claimed in claim 1, wherein the first processor means includes a plurality of
15 registers, and the second processor means has access to a predetermined selection of the said registers.

3. An arrangement as claimed in claim 1 or 2, wherein the first processor means has active and inactive states of operation, and wherein the second
20 processor means is operable to process instructions when the first processor means is in the inactive state.

4. An arrangement as claimed in claim 3, wherein the second processor means is operable to cause the
25 first processor means to change to the active state from the inactive state, when received control signals are indicative of instructions which cannot be processed by the second processor means.

5. A method of operating a computer including
30 first processor means for processing a first set of instructions, and second processor means for processing a second set of instructions, the second set of instructions being a subset of the first set of instructions, the method comprising:

35 using the second processor means to receive

control signals; and

processing instructions in dependence upon the received control signals using the second processor means without reference to the first processor means
5 when the received control signals are part of said second set of instructions.

6. A method as claimed in claim 5, wherein the first processor means has active and inactive states of operation, and instructions are processed using the
10 second processor means when the first processor means is in the inactive state if they are part of said second set.

7. A processing arrangement for a computer, comprising first and second processors, the second
15 processor being code compatible with the first processor, such that instruction code can be shared between the first and second processors and can be operated on by the first or second processor without the need for conversion or alteration.

20 8. A processing arrangement for a computer comprising a plurality of processing elements, each element being assigned a predetermined function, wherein the processing elements are able to share instruction code without conversion or alteration, and
25 the plurality of elements provides functionality equivalent to a predetermined virtual processor.

1 / 3

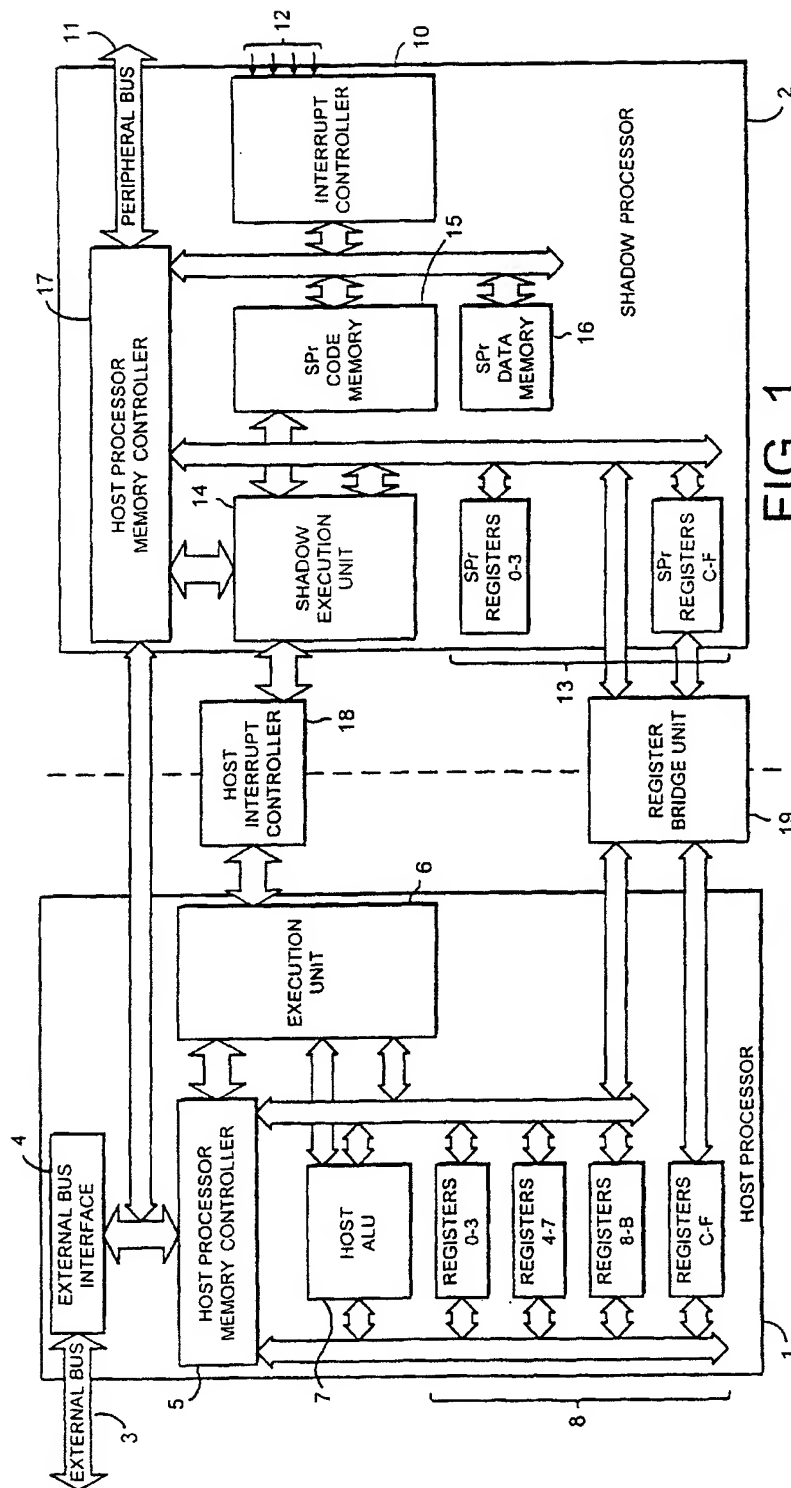


FIG. 1

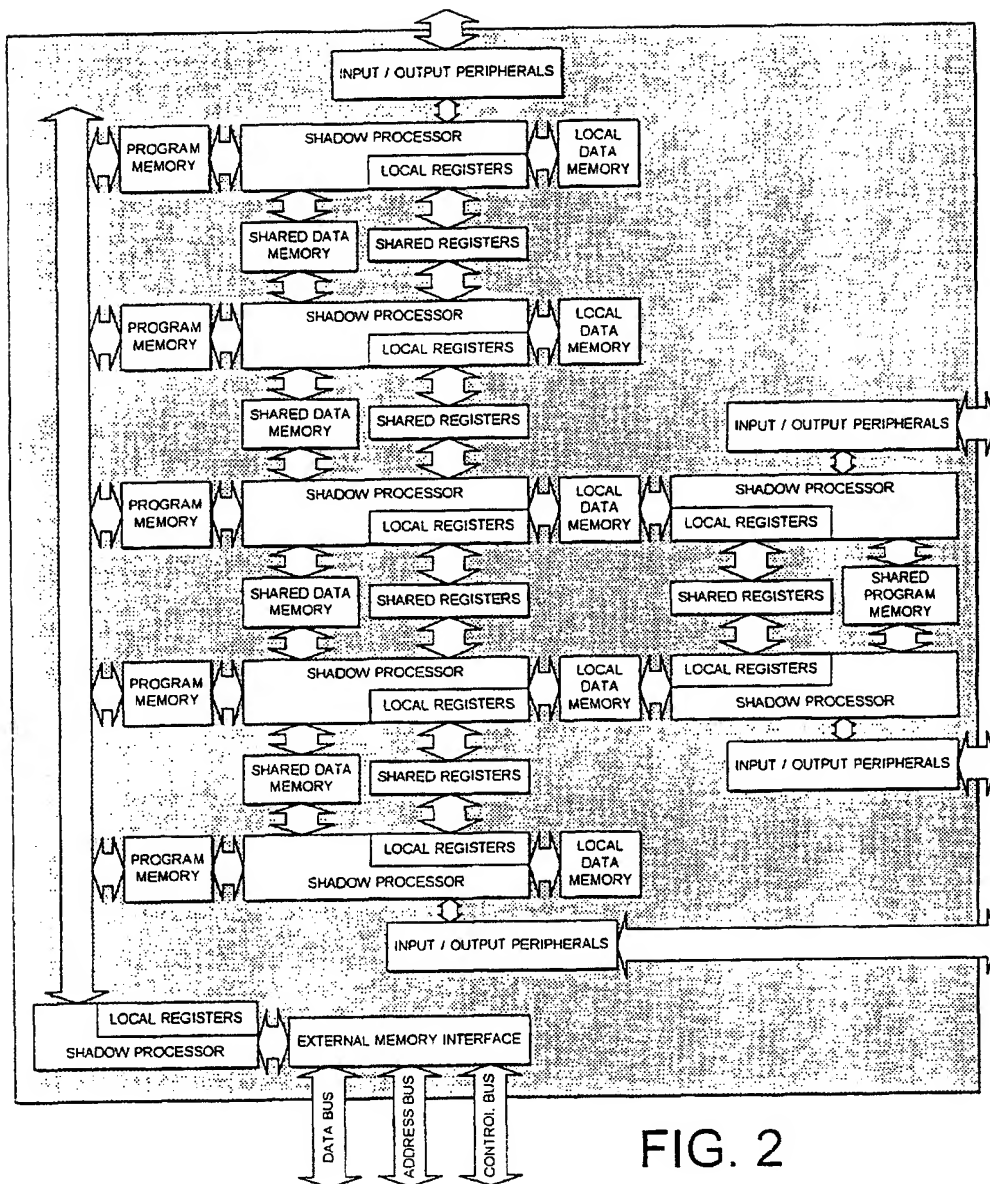


FIG. 2

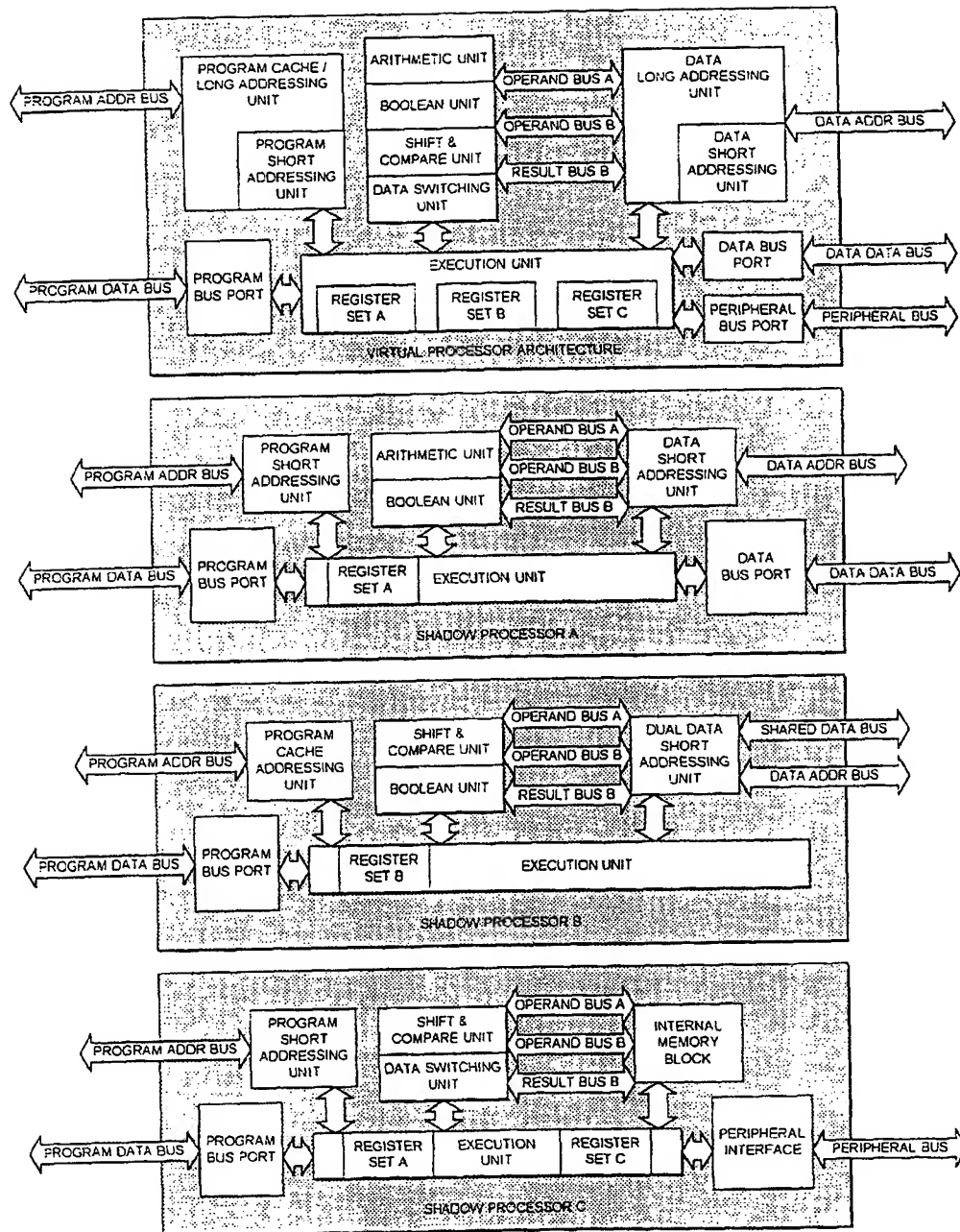


FIG. 3

SUBSTITUTE SHEET (RULE 26)

INTERNATIONAL SEARCH REPORT

Inte. n. Application No

PCT/EP 99/08058

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/38

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 495 588 A (KNOLL KURT ET AL) 27 February 1996 (1996-02-27) column 2, line 35 - column 3, line 18 column 4, line 3 - line 11 column 4, line 56 - line 65 column 5, line 66 - column 6, line 2 ---	1,3-8
X	GB 1 576 276 A (SIEMENS AG) 8 October 1980 (1980-10-08) the whole document ---	1,3,5,7, 8
A	---	4
A	US 5 283 881 A (MARTINI WILLIAM J ET AL) 1 February 1994 (1994-02-01) column 3, line 40 - line 48 ---	2
	--- -/--	

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

31 January 2000

Date of mailing of the international search report

04/02/2000

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Moraiti, M

INTERNATIONAL SEARCH REPORT

Inter national Application No
PCT/EP 99/08058

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No
A	EP 0 174 231 A (TELEMECANIQUE ELECTRIQUE) 12 March 1986 (1986-03-12) page 6, line 5 -page 7, line 34 page 11, line 4 -page 12, line 2 -----	1,3-8

INTERNATIONAL SEARCH REPORT

information on patent family members

International Application No

PCT/EP 99/08058

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5495588 A	27-02-1996	NONE	
GB 1576276 A	08-10-1980	DE 2607685 A	01-09-1977
		AT 353363 B	12-11-1979
		AT 50677 A	15-04-1979
		BE 851840 A	25-08-1977
		CH 610122 A	30-03-1979
		FR 2342530 A	23-09-1977
		IT 1078233 B	08-05-1985
		NL 7701537 A	29-08-1977
US 5283881 A	01-02-1994	NONE	
EP 0174231 A	12-03-1986	US 4716541 A	29-12-1987
		CA 1234223 A	15-03-1988
		CA 1233258 A	23-02-1988
		EP 0174230 A	12-03-1986
		US 4870614 A	26-09-1989